ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

A Unified Framework for Predicting Drugs, Type of Infection, and Side Effects from Disease

[1] Vemuri Saraswathi, [2] Sruthi Yenninti

[1] Department of Computer Science and Engineering, Data Science, Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, India

[2] Assistant Professor, Department of Computer Science and Engineering, Artificial Intelligence & Machine Learning, Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, India

Corresponding Author Email: [1] vemurisaraswathi.23mtech.csd@anits.edu.in, [2] sruthi9814@gmail.com

Abstract*— Understanding disease inquiry has an indispensable role in public health, enabling individuals to catch on to basic health knowledge. As diseases evolve rapidly and new treatments emerge, communication with both drug criteria and medical conditions is becoming critical. This research presents the development of an AI-driven system aimed at intensifying disease perception. To provide strong and reliable data, a custom dataset was curated by assimilating three trusted sources: The National Institute of Health (NIH) pill box retired, The Food and Drug Administration (FDA), and Druglib. For disease exploration, four deep learning models were developed: Multi-task Dense Neural Network, Multi-task Feed forward neural network, Multi-task Gated Recurrent Unit, and Multi-task Hierarchical Attention Network. These models were trained to anticipate the drug name, type of infection, and side effects in conformity with disease input. Among them, the Multi-task Gated Recurrent Unit models achieved a slight maximum performance, with an accuracy of 91.84%. Other performance metrics, such as precision, recall, and confusion matrix were utilized. The overall objective of this research is to develop an artificial intelligence-based unified framework for practitioners to make healthcare decisions based on simplifying knowledge of diseases and drugs, speeding up their diagnosis, and streamlining clinical procedures, thereby ultimately creating better health outcomes.

Index Terms*— Multi-task Dense Neural Network, Multi-task Feed Forward Neural Network, Multi-task Gated Recurrent Unit, Theraplet Recommendations.

I. INTRODUCTION

In today's digital world, a massive amount of data is generated by hospital information systems [1-4], or medical data available online including websites, forums, and journals. Google has emerged as the habitually used reference tool [5], reflecting the prerequisite for accessible and reliable medical information. Publicly accessible datasets like the Food and Drug Administration (FDA) and National Institute of Health (NIH) Pill Box Retired contain immense drug information, where it is difficult for individuals to analyze, interpret, and retrieve specific health details efficiently. This results in data inconsistency, information overload, and non-user friendly. In the same dataset context, it is accommodating complex medical terminology - scientific names for drugs, medical conditions, or diseases where the people feel confused by looking up those terms and unnecessary information like route name, marketing category name, application number, labeler name, created date, update date, etc. The information is not only predominant for the common individual but also for healthcare professionals. This calls attention to the critical gap in public health: that simplifies the process of understanding disease and drug information.

In a way, the world has gradually been altered by the threat of new diseases since 2025. For instance, pathogenic avian influenza A(H5N1) is presently thought to be a potential human-to-human transmissible disease. Drugs resistant to bacteria and viruses are being developed nowadays increasingly imposing constraints on the ability to control diseases such as tuberculosis, malaria, and some viral infections. Instantaneous changes in the infection patterns make it more difficult for healthcare professionals, and the public to be alert and updated with the latest ones. Unlike, the previous researchers mostly worked with symptom-disease and review-drug relationships. In this scenario, the Druglib dataset is widely used, but it does not include dosage and comorbidities. User-friendly interfaces were created but the centre of attention deals with symptoms, exercises, precautions, nearby hospitals, and finally antimicrobial resistance [7].

Dealing with, health care information that encompasses a large volume of data, will be time-consuming with humans but Artificial Intelligence offers a promising solution for exploring medical data. AI can assist by automatically mapping specific diseases, and predicting the most suitable drug, type of infection, and side effects. This not only saves time for healthcare professionals but also reduces medical errors which lead to safer and more accurate healthcare decisions.

To overcome the limitations of the current regime, an AI-based consolidated framework was developed. The configuration identifies the appropriate drug, the type of infection, and side effects with the justification of disease input. For effective implementation of this research, a custom dataset is designed by utilizing three trusted sources: The

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

National Institute of Health (NIH) pill box retired, The Food and Drug Administration (FDA), and Druglib. Four deep learning models: Multi-task Dense Neural Network, Multi-task Feed forward neural network, Multi-task Gated Recurrent Unit, Multi-task Hierarchical Attention Network, and Mini Transformer block – were adopted. The theraplet recommendations mean where a model predicts a single drug, a single side effect, and the type of infection based on the given disease. The main goal of this research is to build an AI-based tool that helps doctors and healthcare workers make better decisions by making disease and drug information easier to understand, speeding up diagnosis, and improving medical processes — all leading to better health outcomes.

II. RELATED WORK

Over the past few years, the accomplishments of Artificial Intelligence in health care procured remarkable attention due to its capabilities to enhance medical decision-making and for efficient solutions. The literature review furnishes an overview of previous work in the field of symptoms-disease, specific disease-drug relationship prediction, and drug information with Health care professionals.

Suvendu Kumar Nayak, et al. [1] developed an intelligent disease prediction and drug recommendation by using multiple approaches of machine learning algorithms. The author organized the research into two modules: (1) Symptom-based Disease prediction and (2) Drug Recommendation based on Disease. To support this framework, the authors created their dataset by merging four datasets that are publicly accessible from the data.World website and UCI Machine Learning Repository. The newly developed dataset contains attributes such as disease name, drug name, reviews, side effects, ratings, occurrence of disease count, and so on. Four machine learning models — the Multinomial Probabilistic Model, ExtraTree Classifier Model, Decision Tree Classifier Model [13,14], and Support Vector Machine Classifier Model [11,13] — were implemented with four different approaches. These models are trained and predict the disease name. On the review attribute, the sentiment analysis is performed with the help of natural language processing. After the forecasting disease, the work advocates the drug recommendations and side effects based on the useful count, sentiment analysis and probabilistic scores. Among the four machine learning models, the Support Vector Machine Classifier Model performed slightly better performance compared to other models.

Isha Santosh, et al. [2] developed a web-based medical application system that utilizes machine learning techniques. In this work, the end user enters the symptoms as an input in text format. The authors exert five algorithms for disease prediction: Random Forest [12,13,15], Support Vector Machine [15,16], K-Nearest neighbors [14], Naïve Bayes [14], and Gradient Boost [13]. After determining the disease, the system provides recommendations such as disease description, precautions, medications, diet, medical centers nearer to the user's location using google maps, and workout. All the models attained an accuracy of 100% based on the input symptoms.

To handle more complex datasets in a real clinical context, Auba Fuster-pala et al. [3] proposed optimized machine learning classifiers for symptom-based disease screening. The author used an open dataset and various models such as a Support Vector Machine, Random Forest, K-nearest neighbor, and Artificial Neural Network. A three-phase enhancement process is used to obtain the best classifier. First dataset is pre-processed, second, a grid search is performed, and finally the best models that are obtained will further undergo an additional filtering process. According to the author's perspective, the foremost model selected based on its performance and execution time is a KNN with 2 neighbors, which achieves an accuracy and f1 score exceeding 98%.

To reduce medical errors, K.P. Swain, S.K. Mohapatra, et.al [8] leveraged natural language processing for patient reviews to progress the drug recommendation system. The system is developed with four main steps: analyzing the review dataset, data pre-processing, model building, and recommending the appropriate drug for a particular disease. Sentiment scores are analyzed by using the VADER (Valence Aware Dictionary and sEntiment Reasoner) and the text data was cleaned using the NLTK (Natural Language Toolkit) library. In medical inquiry, this research presents a novel approach to drug prescription.

To know the utilization of drug information among health care professionals, Abdulkader Hayek, Sathvik B. Sridhar, Syed Arman Rabbani, et. al [5] where a cross-section survey was regulated in eight hospitals (four from public and four from private hospitals), and nine primary healthcare clinics. The collected data is analyzed using the IBM Statistical Package for Social Sciences (SPSS) version 27 software. pharmacists, nurses, and physicians, 311 responses were considered the final analysis.

An innovative approach for antimicrobial resistance using artificial intelligence was devised by Hatim Abdullah Natto, et.al [7] where the AI rapidly scans large chemical databases to predict novel antibacterial compounds. The author referred to the AMR as a Silent Pandemic because of rapid evolutionary changes in bacteria, viruses, and parasites and it is leading to millions of deaths worldwide. Meta SUB AMR is a collaborative AI-based tool for analyzing monitoring and tracking AMR in urban environments. A significant contribution of the paper is the precise and timely identification of new AMR strains, which enables better control over their spread.

Various machine-learning algorithms have been employed in past research for the prediction of disease-drug

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

associations, which follow an indirect pipeline: **symptom->review->disease->drug relationships**. This clearly shows that there is no direct applicability or mapping between the disease and drug where the first analysis starts from the symptom or review which results in the disease name and then to the drug prediction. A summary of the other machine or deep learning models, tasks, datasets used, and the performance compared.

Table 1: Summary of various machine learning or deep learning models

Model	Task	Dataset	Performance
Random-Forest (RF)	Drug-side effect prediction	SIDER, DrugBank	Accuracy ~66%, Precision ~0.69
Logistic Regression (LR)	Side effect prediction from disease	DrugBank, FAERS	Accuracy ~67%, ROC-AUC ~0.72
Support Vector Machine (SVM)	Side Effect Classification or Symptom-Disease Prediction	DrugBank, DrugLib	Accuracy ~70%, Precision ~0.72
Gradient Boost (XG Boost)	Drug-disease link prediction	Custom or OpenFDA	Accuracy ~74%, AUC ~0.78
K-Nearest Neighbours (KNN)	Symptom-Disease Prediction	Custom	Accuracy ~63%
Graph Neural Network (GNN)	Predict side effects using molecule graphs	DrugBank, Decagon	Decagon ROC-AUC ~0.87–0.92
CNN (Text CNN / 1D CNN)	Extract drug-disease relation from text	Drug review text	Accuracy ~88%, F1 ~0.87

The above approaches were carried out based on the symptoms or feedback of drug or molecule graphs to predict the side effects which was time-consuming and more usage of computational resources for prediction and it showed the single predictions based on the given input.

In contrast, the current system perspective was developed to have multiple predictions such as drug name, side effects, and type of infection based on the single input disease name with the help of a curated dataset which shows significant accuracy when compared with the previous methods.

III. PROPOSED SYSTEM

The proposed system is outlined to develop an AI-powered system for disease. It plays a crucial role in supporting both healthcare professionals and the public with fast and efficiency.

A. System Architecture:

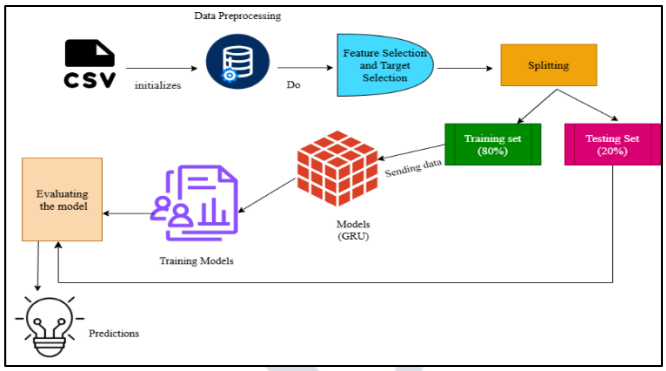

Figure 1. System Architecture for a Unified Deep Learning Framework for Theraplet Recommendations from Disease

a. Data Collection:

Previous studies that are mentioned in the related work where majorly focus on review-based datasets (Druglib) or symptom-based datasets(data.world). In the current study, the centre of attention was shifted to disease-based data curated from a novel dataset with the help of three trusted sources: the National Institute of Health (NIH) Pill Box Retired, Food, and Drug Administration (FDA), and Druglib. The dataset was built up with 7000 rows and 12 columns. The process of building the dataset undergoes several steps as follows:

1. **Disease Collection:** A list of the most frequently occurring diseases was gathered in alphabetical order (A - Z).
2. **Drug Mapping:** Utilizing the drug lib dataset, enumerated drug name was mapped with the respective disease based on the ratings (specifically those of 10/10) and most importantly positive reviews of that drug. On average 1-20 drugs were associated with a disease.
3. **FDA Drug Data Integration:** After tabulating the drug names, other characteristics such as National drug code (NDC), proprietary name (Brand Name), Non-Proprietary name (Generic Name), and Dosage form name were searched every time for each drug.
4. **NIH Drug Properties:** Further attributes such as Active Numerator Strength, Type of Infection, Shape, Color, and Source were accumulated from NIH data.
5. **Additional Fields:** Cover names, commonly used, and side effects are acquired mainly from Google. Side effects were structured to include the most reported reactions linked to each drug name per disease.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

Attribute Information:

Table 2: Summary of the Attributes Disease-Drug Dataset

Attribute Name	Role	Type	Description of Attribute	Missing Values
Brand Name	Feature	Categorical	Name given to the medication by the pharmaceutical company	No
National Drug Code (NDC)	Feature	Integer	The unique 10-digit number assigned in the US by the FDA	No
Generic Name	Feature	Categorical	A chemical name of the drug	No
Dosage Form Name	Feature	Categorical	Pharmaceutical drug products specific form for use	No
Active Numerator Strength	Feature	Integer	Amount of Active Ingredient in the medication, Compared to the total amount of product	No
Type of Infection	Feature	Categorical	Viral, Bacterial, Infectious, Non-Infectious	No
Disease	Feature	Categorical	It is a condition that causes the body or the mind not to work	No
Shape	Feature	Categorical	Physical forms of the medicine like Round, Oval, capsule-shaped, etc.	No
Color	Feature	Categorical	Visible Colour of the drug, tablet-like white, blue, red, yellow, green, pink, etc.	No
Source	Feature	Categorical	Manufactured by the company like Pfizer, Johnson & co., etc	No
Cover Name	Feature	Categorical	The name of the drug on the cover can be a generic name or brand name	No
Commonly Used	Feature	Categorical	The tablets used for a specific disease or condition	No
Side Effects	Feature	Categorical	An unexpected change happens for a medication	No

Note: The NDC used in this dataset is that of the U.S. Food and Drug Administration (FDA). Being authorized codes and openly available, they ensure the dataset's credibility and reliability.

b. Data Preprocessing:

Data preprocessing aims to clean and transform the data into a format that would be suitable for deep learning models. Areas of inconsistency in the data arrangement such as extra space, punctuation, etc. might confuse the models. Since most of the dataset contains categorical data, it was mandatory to convert it into numerical data, as deep learning models typically perform better with numerical inputs. In this scenario, the Python methods and libraries come into existence, for easy conversions. To implement preprocessing the Python methods: Label Encoders, TfidVectorizer, and Tokenizer are employed and libraries: Scikit-learn and Keras are used.

Label Encoders: Label encoders change the categorical data into numerical data which is required for further implementation. First, the label encoder class is instantiated from the sklearn, or scikit-learn library, and the fit_transform method is used to return the encoded labels.

TfidVectorizer: TfidVectorizer is a tool that converts text into weighted numerical features. It is useful for the feature extraction of simpler models. Its output is represented in the sparse matrix.

Tokenizer: It turns the raw text into numerical representations, so the deep learning models understand them. It maps each piece of word into a unique integer ID. These sequenced tokens are input into the models, enabling them to process natural language data effectively.

c. Feature Selection and Target Selection:

Feature Selection is selecting the relevant features from the dataset to improve model performance. By considering the fewer attributes it is often easier to understand and interpret. For instance, the independent variable is the Disease name.

Target Selection involves identifying the variables that must be predicted based on the input features. As it is multi-task, it contains multiple attributes such as drug name, type of infection, and side effects as an output.

d. Splitting:

Dividing the dataset into two or more subgroups is known as splitting. This project for disease inquiry is categorized into two subsets: 80% for training and 20% for testing datasets. Proper splitting of the data from a CSV file is

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

important, especially before feeding the data into various deep-learning models. Training datasets help the models to learn hidden patterns and understand the relationships between the different attributes. Testing datasets are considered as unseen data which is used for evaluation.

e. Deep Learning Models:

Deep Learning models have been a boon for textual data and automatic feature extraction, discerning complex patterns from the text, understanding context, and scaling up large datasets. To effectively predict the drug name, infection types, and potential side effects based on the disease input, four deep-learning models were put into practice and appraised. The models are: Multi-task Gated Recurrent Units, Multi-task Dense Neural Networks, Multi-task Feedforward Neural Networks, and Multi-task Hierarchical Attention Networks.

a. Multi-task Gated Recurrent Unit:

Gated Recurrent Unit is a type of recurrent neural network in deep learning. The main idea behind GRU's is to utilize gating mechanisms which makes it different from the other traditional RNNs. It updates the hidden states at each time step allowing them to remember important information while getting rid of irrelevant details. It aims to simplify the architecture by merging its components. It is effective for sequential data processing and handles multi-output mechanisms. For disease inquiry, it takes the disease name as an input and predicts the drug name, type of infection, and side effects as an output.

b. Model Architecture:

Among the executed models, the Multi-Task Gated Recurrent Unit (GRU) was chosen as the central model due to its higher performance in handling sequential data. It is known for its capability to retain information over long sequences, making it tremendously appropriate for medical data analysis. So, the model architecture for the gated recurrent unit is as follows:

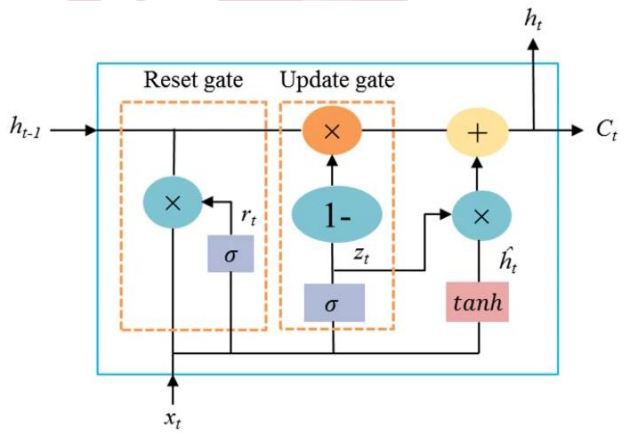

Figure 2. Model Architecture for Gated Recurrent Unit

For building the Multi-task Gated Recurrent Unit model for this module, undergoes various layers:

- Input
- Reset Gate
- Candidate Hidden State
- Update Gate
- Final Hidden State
- Output

Input:

The input is the first where data enters the neural network. The input is the disease name which is in categorical format. It accepts a sequence of integers representing disease names encoded using a tokenizer that is performed in the preprocessing stage. The sequences were padded to maintain equal length for all samples.

Formula:

$$X = [x_1, x_2, \ldots\ldots, x_t], x_t \in \mathrm{Z}$$

Where,

X is thereby a sequence of tokens or word indexes for disease,

Each xt represents the integer ID of a word at time step t (e.g., "headache" – 30, "malaria" – 67, "Aortic Dissection – [18, 23]").

T is the sequence length, which refers to the total number of tokens in each input sentence (or disease name sequence).

The notation $xt \in Z$ means that each token xt is an integer, from the set of all integers Z.

h_{t-1} is Initial Hidden State

The Gated Recurrent Unit is incorporated with two gates and two hidden states. They are as follows:

Reset Gate:

It is the second crucial component gate of the GRU layer. This gate is committed to how much the previous hidden state should be considered or forgotten. Occasionally, the model does not require old information to make a good decision, especially when the new input is relevant. The reset gate helps to ignore irrelevant past information.

Formula:

$$r_t = \sigma\,(W_r\,.\,[h_{t-1}, x_t] + b_r)$$

Where,

W_r is the weight matrix of the reset gate,

h_{t-1} is the previous hidden state,

b_r is the bias.

Interpretation: If $r_t \approx 1$: Remembers the most past state and if $r_t \approx 0$: Forgot most of the past state.

Candidate Hidden State: It is the temporary version of the hidden state, that is calculated based on current input and past information (filtered by reset gate). It represents what the new hidden state could be if the model decides to update it.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

Formula:

$$\tilde{h}t = \tanh(W_h \,.\, [r_t * h_{t-1}, x_t] + b_h)$$

Where,

$\tilde{h}t$ is the candidate's hidden state,

$r_t * h_{t-1}$ is the reset version of the previous hidden state,

x_t is the current input,

W_h is the weight matrix of the operation,

b_h is the bias,

And $tanh$ is the activation function that squashes values between -1 and 1.

Update Gate:

It is the first key component for the GRU layer. This gate decides how much information from the previous state must be carried forward to the next time step. It helps the model to decide whether it should remember past information or update it with new information.

Formula:

$$Z_t = \sigma(W_z \,.\, [h_{t-1}, x_t] + b_z)$$

Where,

σ is the sigmoid activation function, the values lie between 0 and 1,

W_z is the weight matrix for the update gate,

h_{t-1}, is the previous hidden state,

x_t is the current input,

and b_z is the bias.

Interpretation: If $Z_t \approx 1$: Keeps the memory and if $Z_t \approx 0$: Updates the new input.

Final Hidden State:

In a gated recurrent unit, the output layer during the current time step learns all the information that the model knew in its past through both current input and past sequence. This is the combination of:

- The past memory $h_{t-1}$and,
- The new candidate's memory $\tilde{h}$t,
- Decides by updating gate Z_t.

Formula:

$$h_t = (1 - Z_t) * h_{t-1} + Z_t * \tilde{h}t$$

Where,h_t is the final hidden state Z_t is the update gate h_{t-1} is the previous hidden state $\tilde{h}t$ is the candidate's hidden state

Algorithm:

Input:

- Sequence of tokens where the disease name which is encoded into the numerical format is considered as $X = \{x_1, x_2, \ldots\ldots, x_t\}$
- Initialize the hidden state h_0 with zero
- Trainable parameters: In the trainable parameters, the size of the weight matrices can be the input size = n means the length of the word embeddings of the disease names, and hidden size = m. The weight matrices were not manually given by the user, they are randomly initialized.
  1. It contains weight matrices:
     a. Current input (Wz, Wr, Wh),
     b. Previous Hidden States (Uz, Ur, Uh),
     c. biases (bz, br, bh).

Output:

- Final hidden state h_T or all hidden states: Summary of the model $h_T = \{h_1, h_2, \ldots\ldots, h_t\}$

Steps:

1. Start hidden state h_0=0.
2. For every time step t=1 to T:
   a. Compute Reset Gate: $r_t = \sigma(W_r \,.\, [h_{t-1}, x_t] + b_r)$
   b. Compute Candidate Hidden State: $\tilde{h}t = \tanh(W_h \,.\, [r_t * h_{t-1}, x_t] + b_h)$
   c. c. Compute Update gate: $Z_t = \sigma(W_z \,.\, [h_{t-1}, x_t] + b_z)$
   d. d. Compute the Final hidden state: $h_t = (1 - Z_t) * h_{t-1} + Z_t * \tilde{h}t$
3. Return h_T or the full sequence: Summary of the model $\{h_1, h_2, \ldots\ldots, h_t\}$

Dropout Layer:

It is a regularization technique, which involves randomly dropping the neurons and prevents overfitting during training. It is applied to the input connections. 30% of the neurons are removed from the layer. It makes the model more robust by encouraging it to learn distributed representations.

Dense Layer:

A dense layer is a layer where every input neuron is connected to each output neuron. It takes the output from of the GRU + Dropout layer and learns the weights to the predicted output.

$$h_{dense} = ReLU(W_d \,.\, \tilde{h}T + b_d)$$

$\tilde{h}$T: Output from the Dropout layer from the GRU, which is a vector of length 128 (as GRU (128)).

W_d: The dense layer's weight matrix that transforms input from 128 to 64 dimensions. Shape: $W_d \in$ R 64×128.

b_d: The dense layer's bias vector. It provides a trainable offset to every output unit. Shape: $b_d \in$ R 64·The standard matrix multiplication of weights over input.

h_{dense}: The output vector from the dense layer, is fed into output layers. Shape: R64

Output Layer:

Each output layer is mapped to each dense layer with 64 units or neurons which captures relevant features. The final output for each task (Drug name, Type of infection, and Side Effects) is generated by the SoftMax activation function applied to encoded labels.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

Formula:

$$\hat{y}i = Softmax(W_i \ . h + b_i), for\ i \in \{drug, infection, side\}$$

$\hat{y}$i: Predicted output for the task i (drug, infection, side)

W_i: Weight matrix of specific task of dense layers (drug, infection, side)

h: Final hidden states

b_i: Bias vector for task i

SoftMax Activation Function:

It turns the list of numbers into probabilities. It is also used for the multi-output classification. For a given score Z_i among all scores $Z_1, Z_2, Z_3, \ldots\ldots, Z_n$.

Formula:

$$Softmax = \frac{e^{Z_i}}{\sum_{j=1}^{K} e^{Z_j}}$$

Where,

Z_i is the score for class i,

K is the total number of classes.

c. Multi-task Dense Neural Network:

Each layer is linked to the preceding layer. It consists of one or more layers, where they are associated consecutively. It has successive layers: **Input layer:** It is the first layer of the neural network. It does not perform any mathematical calculations, and it just progresses to the upcoming layers. **Hidden Layer:** The weights were rearranged at the time of training to learn patterns. In disease inquiry, three hidden layers were utilized. The foremost features were comprehended first with 128 units. The second layer reduces the neurons which have 64 units and makes the model more efficient. For best predictions, a third layer is employed. **Output Layer:** It is the last layer of the dense neural network. Three dense layers were used for the output prediction of drug name, type of infection, and side effects. The activation function is SoftMax.

Formula for hidden layer:

$$h^{(l)} = ReLU(W^{(l)}\ h^{(l-1)} + b^{(l)})$$

$h^{(l)}$: Output of the lth hidden layer, $W^{(l)}$: Weight matrix of the lth layer, $h^{(l-1)}$: Output of the previous layer, $b^{(l)}$: Bias Vector for the lth layer

d. Multi-task Feed Forward Neural Network:

It is the simplest and first neural network in deep learning models. Each layer was connected by weights. **Input Layer:** It takes the input data of disease which is in the numerical format. **Hidden Layer:** The vectorized data is taken from the input and passed to the further hidden layers. In hidden layer, it has two components: Forward propagation and Background Propagation. In Forward Propagation, the output from the input is based on the summation of weights, activation functions, and evaluating the loss. In Backward propagation, the loss is reduced, the weights, and the models for better predictions. **Output Layer:** It is the finishing layer of the network where each output has a separate dense layer for predictions of drug name, type of infection, and side effects.

e. Multi-task Hierarchical Attention Network:

Hierarchical Attention Networks are important in deep learning. It allows to understand the sequential data also when it is hierarchically structured. **Input Layer:** Takes in a sequence of word indexes from the tokenized text of a disease. **Embedding Layer:** Converts tokens into 128-dimensional vectors. **Word Encoder:** It processes each word in the data using a neural network, which can utilize BiLSTM.

Formula:

$$h_t = BiLSTM(e_t) = [\vec{ht}, \overleftarrow{ht}]$$

Where,

e_t : This is the vector for the word at time step t, $BiLSTM(e_t)$ Here, LSTM is applied in both directions: Forward LSTM (i.e., from left to right): $\vec{ht}$, Backward LSTM (right to left): $\overleftarrow{ht}$, Finally, the output is ht = $[\vec{ht}, \overleftarrow{ht}]$

Word Attention: It focuses on the most important words of the numerical sequential data. In this research, applying the custom attention layer for the data which helps for more attention on the relevant attribute. It breaks down into 3 parts: Compute Intermediate Attention Scores, Compute Attention Weights, and Weighted Sum of Hidden States.

Formulas:

Compute Intermediate Attention Scores:

$$u_t = \tanh(W\ .\ h_t + b)$$

Compute Attention Weights:

$$\alpha_t = \frac{e^{u_t^T u}}{\sum_{t`} e^{u_t^T u}}$$

Compute the Weighted sum of hidden states:

$$s = \sum \alpha_t\ h_t$$

Sentence Encoder: It takes the input from the previous work attention step and runs through the neural network. With the help of a sentence encoder, obtain a large matrix of sentence representations, which are then fed into the Sentence Attention layer. **Sentence Attention:** Sentence attention assigns different weights to each word, concentrating on important sentence-level information. **Fully connected Layers:** with a dense layer that helps the model identify patterns across the data before making the final prediction, along with a dropout layer to help prevent overfitting. The word-to-word attention output comes in the form of sentence attention, indicated by s. This then gets fed to a Dense (fully connected) layer with ReLU activation.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

Dense layer formula:

$$h = ReLU(W_d . s + b_d)$$

s: This is the attention-weighted sentence representation from the previous layer. **Output Layer:** Each output layer is mapped with a dense layer and encoders of drugs, infection types, and side effects are also passed with the activation function of SoftMax.

f. Mini Transformer Block:

To process the sequential textual data and provide more attention to each class for efficient processing. In the mini transformer block, first, the input was converted into dense vectors. Second, the self-attention mechanism was applied to all words in the sequence that were extracted from the CSV file with the help of the. values method. It is mathematically represented as follows:

$$Attention(Q, K, V) = softmax\left(\frac{QK}{\sqrt{d_k}}\right)V$$

Later the attention layer was added with the normalization where the input across the feature samples independently. It is represented as

$$x = LayerNorm(x + MultiHeadAttention(x)$$

Using the Feed feed-forward neural Network the weight and biases were applied to dense layers with RELU activation:

$$FFN(x) = RELU(xW_1 + b_1)W_2 + b_2$$

Finally, employing the GlobalAveragePooling2D converts the three-dimensional vector into a two-dimensional vector by reducing the number of parameters.

f. Training and evaluating the model, and predictions:

The dictionary of target labels for each output layer such as drug name, type of infection, and side effects are assigned with key names such as drug, infection, and side. A fit function is used to train the model. The model learns the patterns from training data. The testing data is used to evaluate the performance of the model, as it contains unseen data that is not included in the training. Each attribute is trained simultaneously, to improve overall efficiency.

Sample Predictions for the disease inquiry:

Input: Giving the disease name as input for the deep learning model.

```
Disease Name: Cholera
```

Figure 3. Disease Inquiry Input

Output:

```
Disease Name: Cholera
Predicted Drug: Cefixime
Predicted Infection Type: Bacterial infection
Predicted Side Effects: Tooth discoloration
```

Figure 4. Predicted Outputs (Drug, Infection Type, Side Effects)

IV. RESULTS

Employed the Multi-Task Gated Recurrent Unit, Multi-Task Feed Forward Neural Networks, Multi-Task Dense Neural Network, Multi-Task Hierarchical Attention Network, and Multi-Task Mini Transformer block to forecast the drug name, type of infection, and side effects. Multi-task Gated Recurrent model achieved the maximum performance, with an accuracy of 91.84%, precision of 92.50%, and recall of 91.84%. Even though the Multi-Task Mini Transformer reached the GRU, in terms of model architecture, implementation, and sequential handling GRU performed well. The result shows the final verdict from the simplest to complex models. The performance metrics of all deep learning models that are considered are accuracy, Precision, Recall, and confusion matrix. For each performance metric, one output class was considered to represent the values.

A. Accuracy:

It is the correspondence of several correct predictions and a total number of predictions.

Formula:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} = \frac{TP+TN}{TP+TN+FP+FN}$$

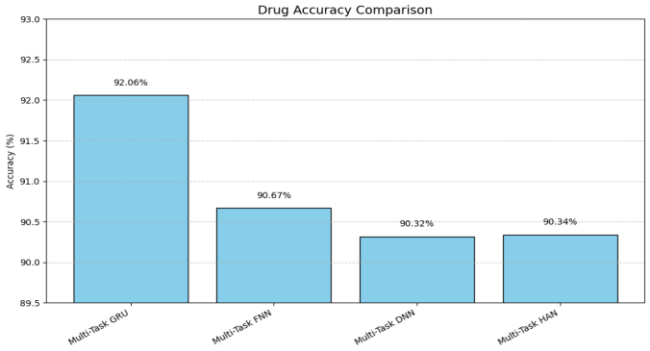

Figure 5. Model Accuracy of Various deep learning

Figure_5 considers the drug accuracy deep learning models such as Gated Recurrent Units (92.06), Dense Neural Networks (90.32), Feed-forward Neural Networks (90.67), Hierarchical Attention Networks (90.34), and Mini Transformer Block (91.33).

B. Recall:

It is one of the performance metrics where the actual positive values are classified correctly.

Formula:

$$Recall = \frac{TP}{TP + FN}$$

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

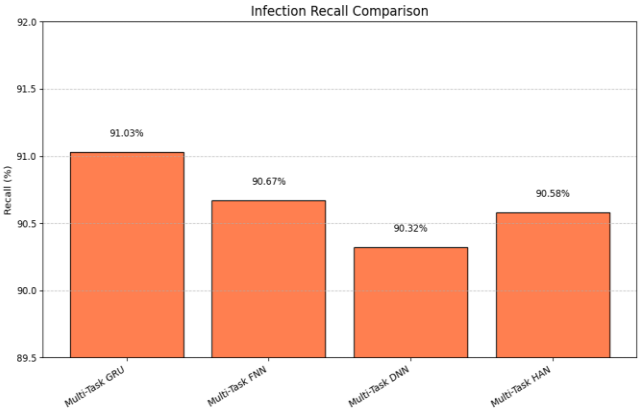

Figure 6. Model Recall of Various Deep Learning

Figure_6 shows only the type of infection recall deep learning models such as Gated Recurrent Units (91.03), Dense Neural Networks (90.32), Feed-forward Neural Networks (90.67), and Hierarchical Attention Networks (90.58), and Mini Transformer block (91.01).

C. Precision:

It is used to compute how many positive classes are correctly identified.

Formula:

$$Precision = \frac{TP}{TP + FP}$$

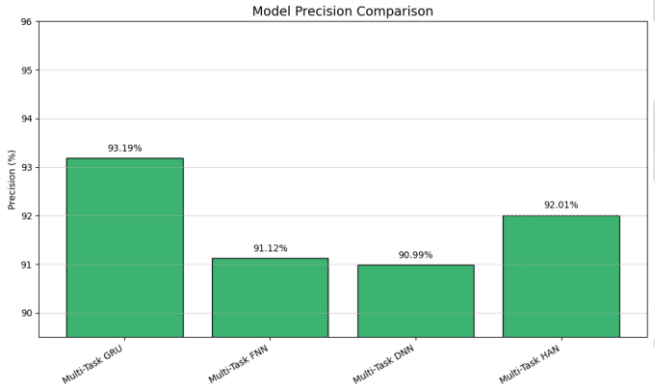

Figure 7. Side Effects Precision of Various Deep Learning

Figure_7 shows only the side effects of precision deep learning models such as Gated Recurrent Units (93.19), Dense Neural Networks (91.12), Feed-forward Neural Networks (90.99), Hierarchical Attention Networks, and Mini Transformer Block (93.01).

Where,

As this is a multi-task output, there are three output classes Drug name, type of infection, and side effect. Each class is computed with the metric terms such as True Positive, True Negative, False Positive, and False Negative.

True Positive (TP): The model-predicted class correctly predicts the correct actual label. For example, if considering the drug class for the disease tuberculosis (the model is predicted Pyrazinamide), then the actual label is also a Pyrazinamide.

True Negative (TN): Both the predicted class and actual label are incorrectly predicted.

False Positive (FP): The model predicts correctly but it does not match with the true label.

False Negative (FN): The model predicts incorrectly, which does not match with the true label.

Table 3: Performance metrics of the deep learning models in the proposed system

S.No	Algorithm	Accuracy	Precision	Recall
1.	Multi-task Gated Recurrent Unit	91.841	92.5075	91.841
2	Multi-Task Dense Neural Network	90.326	90.1870	90.326
3	Multi-Task Feed Forward Neural Network	90.676	91.117	90.676
4	Multi-Task Hierarchical Attention Network	90.123	90.6270	90.1640

Table 3 shows the total accuracy, total precision, and total recall by summing up the individual values of the drug, type of infection, and side effects of each performance metric.

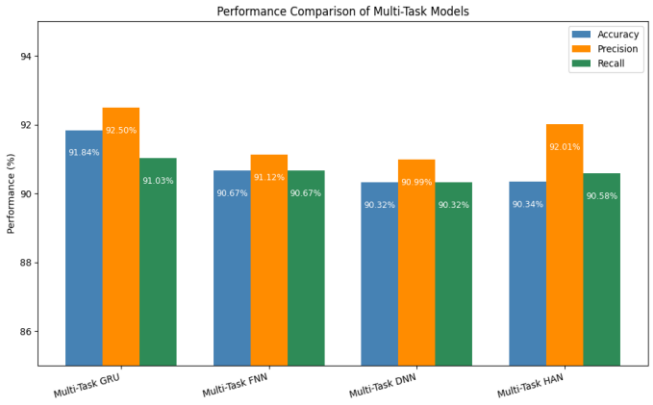

Figure 8. Performance Comparison for deep learning models

Figure_8 shows all the performance metrics that were shown in table 3 of the result section.

D. Confusion matrix:

To check how well the model classification is accomplished by differentiating between the actual values and predicted values.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 06, June 2025

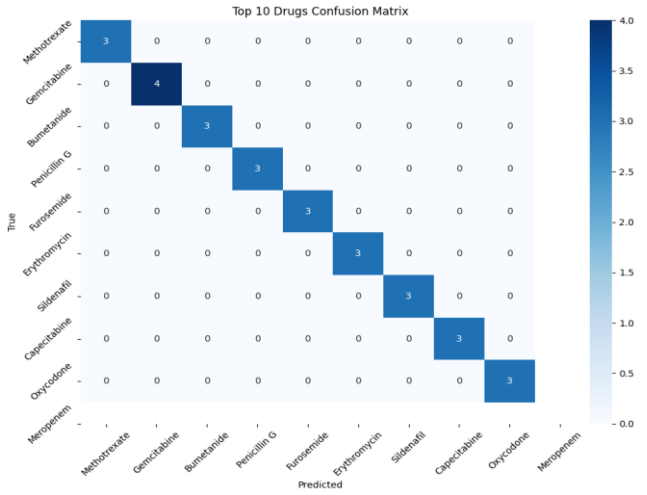

Figure 9. Confusion matrix for the top 10 predicted drugs based on disease

Figure_9 shows that the confusion matrix represents the relationship between the disease with the drug name. The zero clearly states that a specific drug is not misclassified by another drug. For example, [Methotrexate, Gemcitabine] means that Methotrexate was never mistakenly predicted as Gemcitabine.

V. DISCUSSION

Previous works enhance the single-output predictions based on the given input, typically symptom or drug review, or molecular structure, where the accuracies show the range between a range of 60 – 80+ shown in Table 1(Related work). All the deep learning models in the proposed system attain an accuracy exceeding 90%, even if it is multi-output (Drug, Side Effect, and Type of Infection) by taking a single input (disease name).

VI. CONCLUSION

The robust Unified Framework for Predicting Drugs, Infections, and Side Effects Using Deep Learning presented in this research incorporates disease-based drug prediction, aimed at improving and accelerating the decision-making process in a healthcare setting. By integrating advanced deep learning models—namely, Multi-Task GRU, DNN, FNN, HAN, and mini transformer block—this system enables accurate prediction of effective drugs, infection types, and potential side effects.

In addition, The Multi-Task GRU delivers maximum performance at 91.84 percent accuracy, thus proving the potential of recurrent architectures in capturing intricate temporal characteristics found in many medical datasets.

In this way, the system addresses the problem of resistance to drug therapies and provides well-grounded future avenues for personalized treatment recommendations. Notably, this research aligns the traditional healthcare-centric data storage models with intelligent automation for the real-world applicability of AI-inspired solutions. Further developmental strides may add multilingual features, EHR integration, and voice-assisted interfaces to the system so that its use becomes more widely available and applicable in different clinical environments. Overall, this pronged AI-based framework stands as a great achievement toward safer, faster, and more informed clinical practices.

REFERENCES

[1] Suvendu Kumar Nayak, Mamata Garanayak, Sangram Keshari Swain, Sandeep Kumar Panda, and Deepthi Godavarthi “An intelligent disease prediction and drug recommendation prototype by using multiple approaches of machine learning algorithms”, Institute of Electrical and Electronics Engineers, 2023.

[2] Auba Fuster-Palà, Francisco Luna-Perejón, Lourdes Miró-Amarante, and Manuel Domínguez-Morales, “Optimized Machine Learning Classifiers for Symptom-Based Disease Screening”, Multidisciplinary Digital Publishing Institute, 2024.

[3] Fatemeh Sogandi “Identifying diseases symptoms and general rules using supervised and unsupervised machine learning”, Scientific Reports, 2024.

[4] Harikrishnan.M, Viji Vinod, V. Sarala Devi, “SYMPTOM BASED DISEASE PREDICTION USING MACHINE LEARNING AND TKINTER”, International Research Journal of Modernization in Engineering Technology and Science, 2024.

[5] Abdulkader Hayek, Sathvik B. Sridhar, Syed Arman Rabbani, Javed Shareef, “From textbooks to online sources: An Investigation of drug information resource preference and utilization among healthcare Professionals”, Clinical Epidemiology and Global Health, 2024.

[6] K.P. Swain, S.K. Mohapatra Vinayakumar Ravi, Soumya Ranjan Nayak, Tahani Jaser Alahmadi, Prabhishek Singh, and Manoj Diwakar, “Leveraging Machine Learning and Patient Reviews for Developing a Drug Recommendation System to Reduce Medical Errors”, Open Bioinformatics Journal, 2024.

[7] Hatim Abdullah Natto, Ammar Abdul Razzak Mahmood, Sriram Thiruvengadam, Rajkumar Krishnan Vasanthi, Desh Nidhi Singh, “Artificial intelligence in combating antimicrobial resistance”, IP International Journal of Medical Microbiology and Tropical Diseases Official Publication of Khyati Education and Research Foundation, 2024.

[8] Manikanta Sirigineedi1, Matta Eswar Surya Manikanta Kumar, Rali Surya Prakash, Velagala Pavan Kumar Reddy, Poojitha Tirunagari, “Symptom-Based Disease Prediction: A Machine Learning Approach”, Journal of Artificial Intelligence, Machine Learning and Neural Network, 2024.

[9] Etana Fikadu Dinsa, Mrinal Das, Teklu Urgessa Abebe, "AI-based disease category prediction model using symptoms", Scientific Reports, 2024.

[10] Devansh Ajmera, Trilok Pandey, "Early-Stage Disease Prediction from Various Symptoms Using Machine Learning Models" Research Gate,2024.

[11] K. Gaurav, A. Kumar, P. Singh, A. Kumari, M. Kasar*, T. Suryawanshi, "Human Disease Prediction using Machine Learning Techniques and Real-life Parameters", International Journal of Engineering, 2023.

[12] Palle Pramod Reddy, Dirisinala Madhu Babu, Hardeep Kumar and Dr. Shivi Sharma, "Disease Prediction using Machine Learning", International Journal of Creative Research Thoughts, 2021.

[13] C. Kavitha, Vinodhini Mani, S.R. Srividhya, Osamah Ibrahim Khalaf, Carlos Andres Tavera Romero, "Early-Stage Alzheimer's Disease Prediction Using Machine Learning Models", frontiers, 2022.

[14] Laxmi Deepthi Gopisetti, Srinivas Karthik Lambavai Kummera, Sai Rohan Pattamsetti, Sneha Kuna, Niharika Parsi, Hari Priya Kodali, "Multiple Disease Prediction System using Machine Learning and Streamlit", Institute of Electrical and Electronics Engineers, 2023.

[15] Supriya Telsang, Tanvi Pokale, Anagha Posugade, Pradyot Patwal, Prachi Shirode, Om Popade, "Heart Disease Prediction using Machine Learning", Institute of Electrical and Electronics Engineers, 2024.

[16] Swaroop Sana, "MULTI DISEASE PREDICTION SYSTEM USING MACHINE LEARNING", ResearchGate, 2024.

[17] Ridham Sood, Virat Sharma, "Symptom Based Disease Prediction System using Machine Learning", International Journal of Preventive Medicine and Health, 2024.

[18] Marouane Ferjani, "Disease Prediction using Machine Learning", ResearchGate, 2020.

[19] Aashish Gnanavelu, Champa Venkata ramu and Rama krishna Chintakunta, "Cardiovascular Disease Prediction Using Machine Learning Metrics", Journal of Young Pharmacists,2025.

[20] Rakibul Islam, Azrin Sultana, Mohammad Rashedul Islam, "A comprehensive review for chronic disease prediction using machine learning algorithms", 2024.